

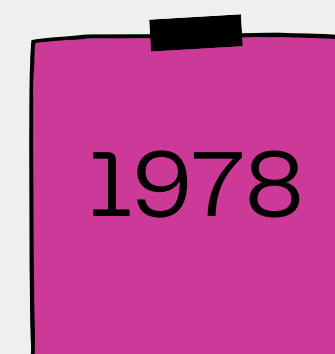
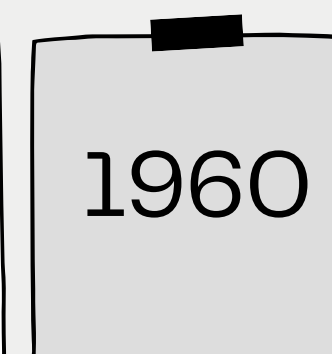
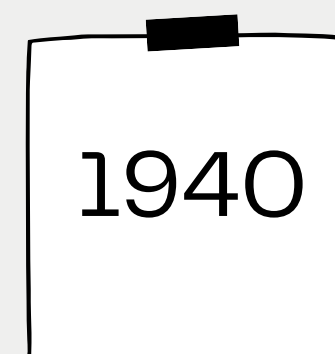
Programação Funcional

História

Introduzida pelo pensamento do matemático Alonzo Church, a base do pensamento funcional é proveniente do fundamento da computação do cálculo- λ .

1

Processo;



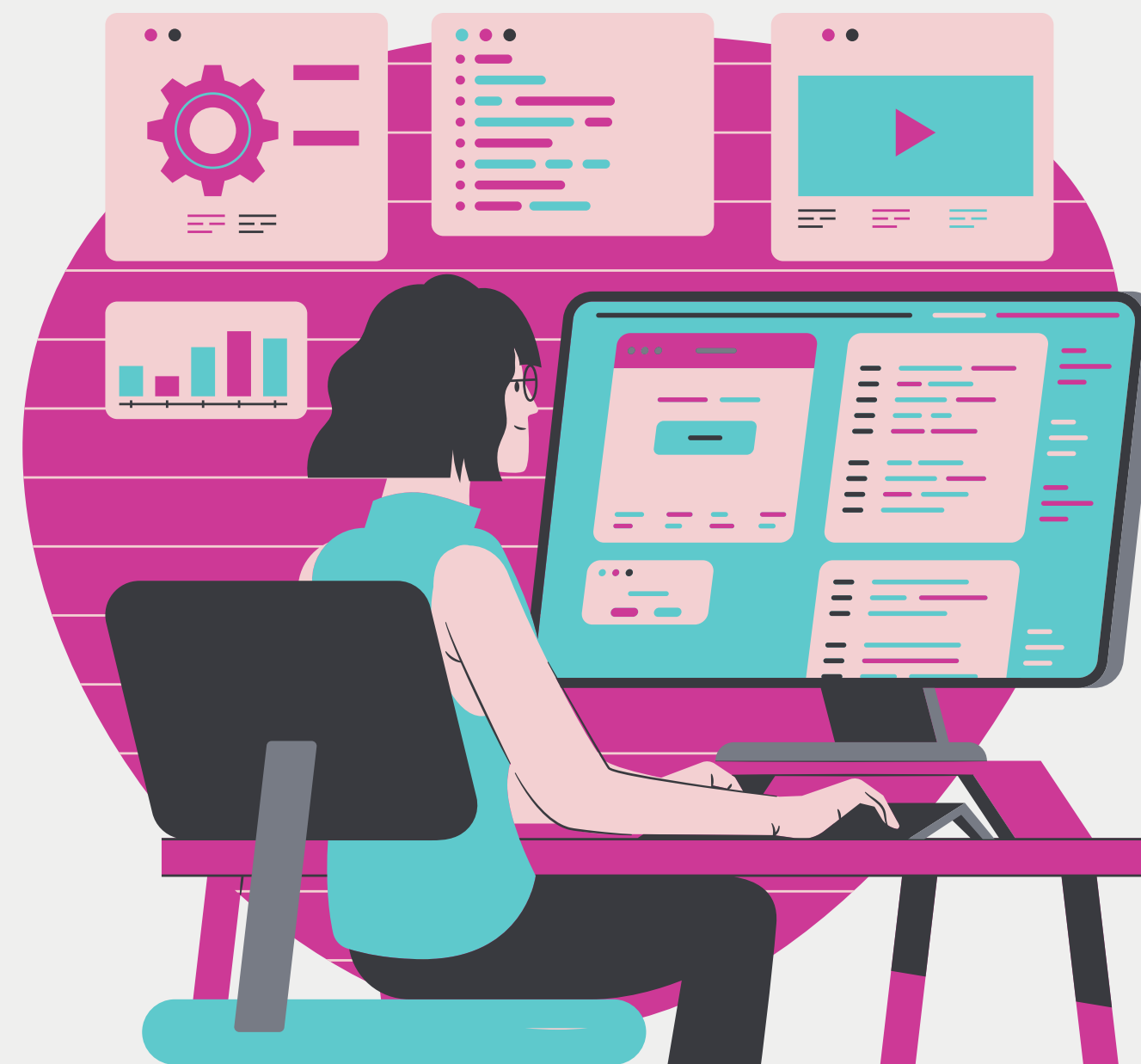
2

Atualidade;

Paradigma

Estabelecendo um cenário adequado para o projeto, os paradigmas, além de uniformizar o trabalho, proporcionam também otimização através da metodologia modelada por cada estilo de estruturação.

- 1 Estruturas de códigos;
- 2 Aplicação variável específica;
- 3 Objetivo final;



Q Paradigma Funcional

1

Uso de Funções;

3

Imutabilidade de variáveis;

2

Estruturação em blocos;

4

Aplicações e Benefícios;

Q Paradigma Funcional: Uso de Funções

https://github.com/ikimychan/Paradigma_Funcional/wiki

Benefícios do uso de funções – Wiki

11 hours ago – As funções definem uma estrutura sólida, fazendo com que sejam otimizadas por serem compostas por um código limpo, facilitando o processo de manutenção.

https://github.com/ikimychan/Paradigma_Funcional/wiki

Tipos de função – Wiki

10 hours ago – Existem alguns tipos de função que se destacam quando este paradigma é utilizado. Estas são caracterizadas como: Puras, de primeira classe e as "High order".

Paradigma Funcional: Uso de Funções


https://github.com/ikimychan/Paradigma_Funcional/wiki

Benefícios do uso de funções – Wiki

11 hours ago – As funções definem uma estrutura sólida, fazendo com que sejam otimizadas por serem compostas por um código limpo, facilitando o processo de manutenção.

https://github.com/ikimychan/Paradigma_Funcional/wiki

Tipos de função – Wiki

10 hours ago – Existe  alguns tipos de função que se destacam quando este paradigma é utilizado. Estas são caracterizadas como: Puras, de primeira classe e as "High order".

Q Paradigma Funcional: Uso de Funções

https://github.com/ikimychan/Paradigma_Funcional/wiki

Benefícios do uso de funções – Wiki

11 hours ago – As funções definem uma estrutura sólida, fazendo com que sejam otimizadas por serem compostas por um código limpo, facilitando o processo de manutenção.

https://github.com/ikimychan/Paradigma_Funcional/wiki

Tipos de função – Wiki


10 hours ago – Existem alguns tipos de função que se destacam quando este paradigma é utilizado. Estas são caracterizadas como: Puras, de primeira classe e as "High order".

🔍 Paradigma Funcional: Tipos de função

← → 🏠

https://github.com/ikimychan/Paradigma_Funcional/wiki

☆ ☰



Tipos de função – Wiki

Wiki – Paradigma: Programação funcional.

Funções Puras

Resultados que são exclusivamente dependentes do que já foi inserido nos parâmetros. Estas proporcionam simplicidade arquitetural para o código, evitando loops.

Funções de Primeira Classe

Convertem funções em dados. Desta forma, as funções criadas como de primeira classe tem valores que podem ser realocados, manipulados e retornados por outros comandos no código.

Funções High Order

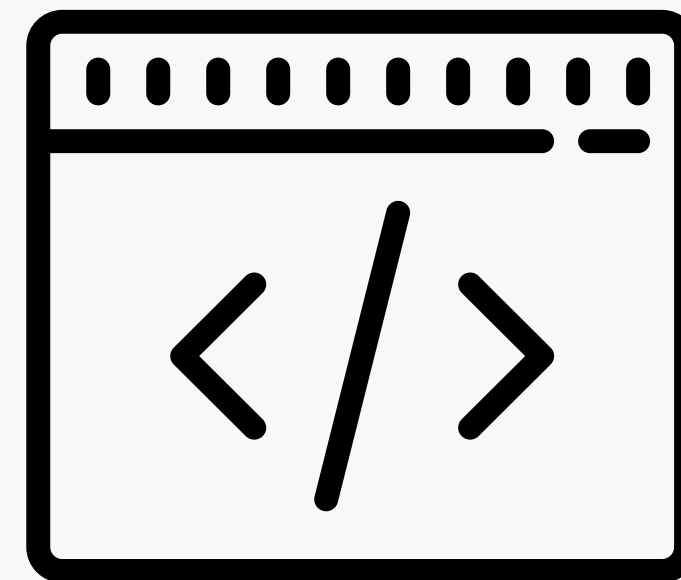
Responsáveis por receber uma função qualquer e retornar outra função, sendo que esta utiliza uma ou mais funções como argumento.

Estruturação

A programação funcional tem como base a estruturação feita em blocos.

Os benefícios da formação em blocos consistem principalmente na facilidade de localização, otimizando os processos de manutenção e se adequando para projetos grandes.

Q Estrutura em Blocos



Estruturação

A programação funcional tem como base a estruturação feita em blocos.

Os benefícios da formação em blocos consistem principalmente na facilidade de localização, otimizando os processos de manutenção e se adequando para projetos grandes.

Q Orientada a Objeto

```
class Nome {  
  constructor(nome, familia) {  
    this.nome = nome;  
    this.familia = familia;  
  }  
  getDetails(){  
    return (` O nome completo é ${this.nome}  
    ${this.familia} .` )  
  }  
}  
  
let pessoa1 = new Nome('Lucas', 'Angelim');  
  
console.log(pessoa1.getDetails());
```

Estruturação

A programação funcional tem como base a estruturação feita em blocos.

Os benefícios da formação em blocos consistem principalmente na facilidade de localização, otimizando os processos de manutenção e se adequando para projetos grandes.

Q Funcional

```
var GeradorNomes = function  
(nome, familia) {  
    console.log("O nome completo  
é", nome, familia);  
};
```

```
let pessoal = new  
GeradorNomes("Maria", "Kimura");  
console.log();
```



Imutabilidade

As variáveis não são alterados fora de suas funções para que não existam efeitos colaterais. Tornando o processo facilmente aplicável a sistemas extensos e complexos.

Funcional em prática

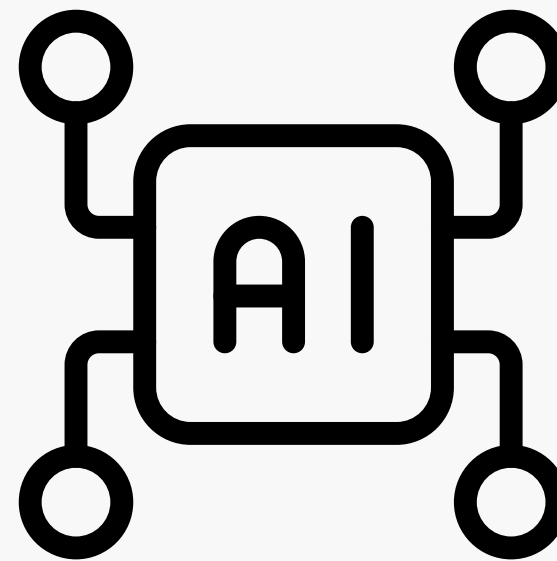
1 Vantagem;

Confiabilidade e estabilidade proporcionada pelo código estruturado em blocos, possuindo busca simplificada por eventuais erros e bugs, permitindo com que essa seja empregada em sistemas de larga escala.

2 Desvantagem;

Desempenho mais lento para processamento e para sua construção. Manutenção dificultada por objetos que sofrem evoluções ou valores imutáveis. O código também pode ser dificultado para iniciantes.

Q IA



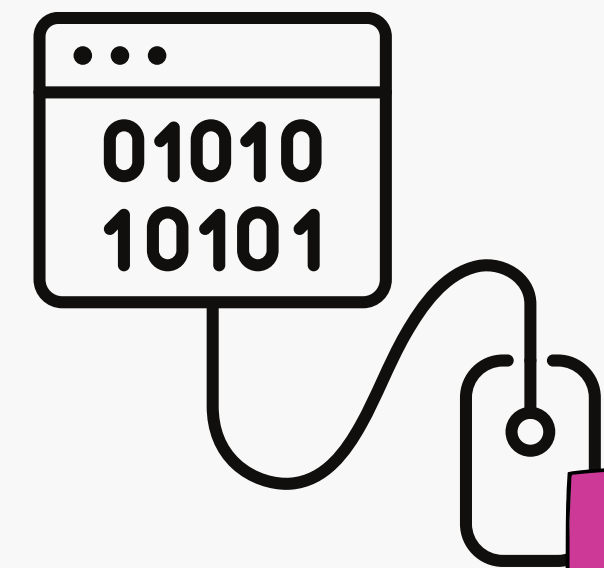
Q Big Data



Q Análise Estatística



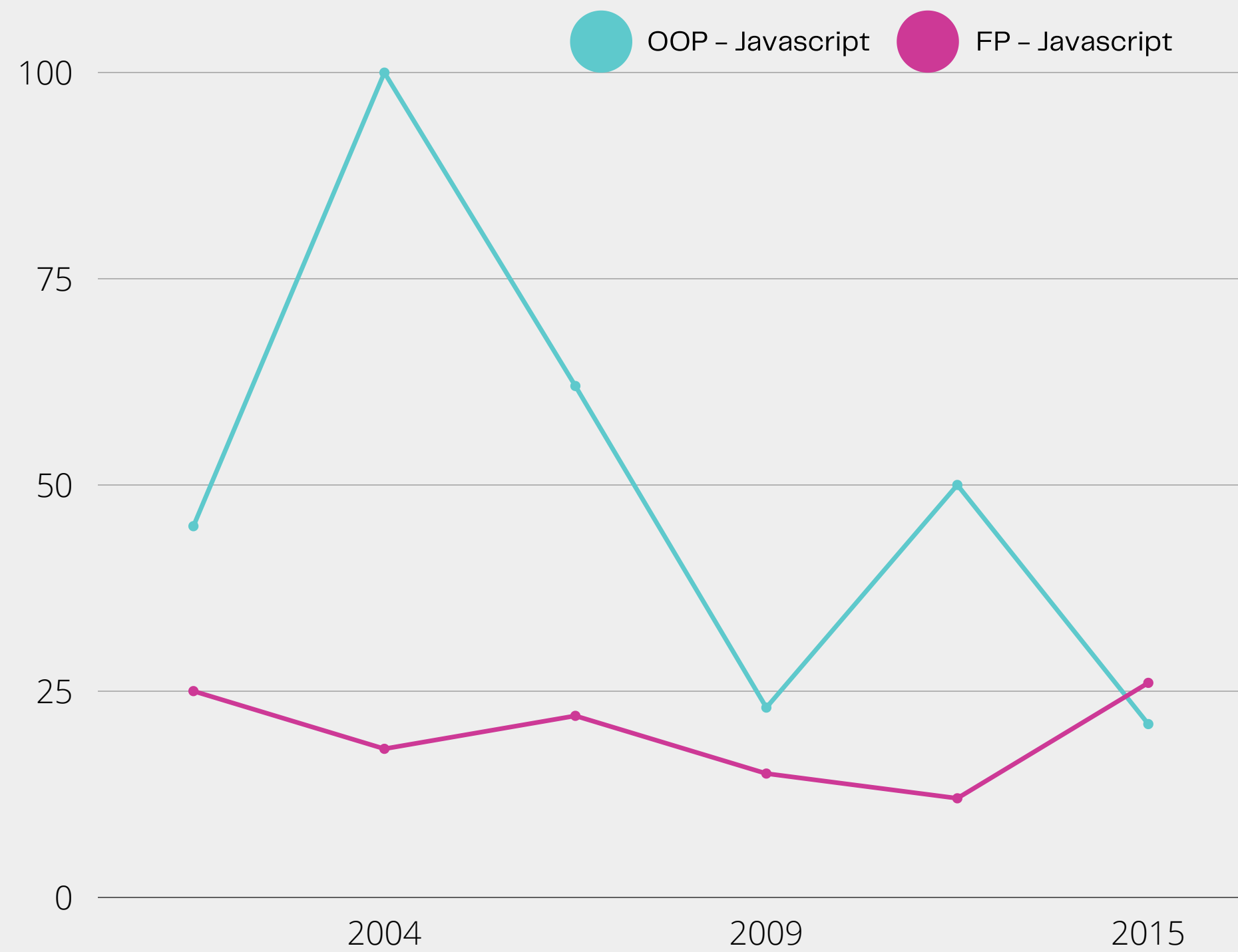
Q Uniformização



Funcional
vs
Imperativas?



Paradigma Funcional: Popularização



Google Trends - Interest over time.

Funcional = Mindset

Organização do código e do pensamento.

Q Funcional se torna sinônimo de ordem

Os paradigmas funcionais estabelecem um paralelo interessante entre o comportamento do programador e seu modo de pensar e aquilo que é inserido em seu código.

Uma vez que o programador possui um planejamento e pensamento organizado para estruturar seu código, a implementação do método funcional se torna ágil.

Sendo possível adquirir a organização como característica, otimizando também, o programador em sua realização pessoal.





Programadores não programam para máquinas,
programadores programam para outros
programadores possam desenvolver.

The poetry of programming | Linda Liukas | TEDxCERN

Agradecemos pela
atenção